

Results for "(((freeze list) or snapshot) &lt;sentence&gt; ((quiesce strength) or (level &lt;near/3&gt; risk &lt;..."))"



Your search matched 6 of 1193303 documents.

A maximum of 100 results are displayed, 25 to a page, sorted by Relevance in Descending order.

[View Session History](#)
[New Search](#)

Modify Search

» Key

☒ IEEE JNL IEEE Journal or Magazine

☐ Check to search only within this results set

☐ IEE JNL IEE Journal or Magazine

Display Format: ☒ Citation ☐ Citation & Abstract

☐ IEEE CNF IEEE Conference Proceeding

Select Article Information

☐ IEE CNF IEE Conference Proceeding

- ☐ 1. **Efficient detection and resolution of generalized distributed deadlocks**  
 Kshemkalyani, A.D.; Singhal, M.;  
 Software Engineering, IEEE Transactions on  
 Volume 20, Issue 1, Jan. 1994 Page(s):43 - 54  
[AbstractPlus](#) | Full Text: [PDF\(984 KB\)](#) IEEE JNL
- ☐ 2. **Fault-tolerant distributed simulation**  
 Groselj, B.;  
 Simulation Conference, 1991. Proceedings., Winter  
 8-11 Dec. 1991 Page(s):637 - 641  
[AbstractPlus](#) | Full Text: [PDF\(360 KB\)](#) IEEE CNF
- ☐ 3. **Optimal deadlock detection in distributed systems based on locally constructed**  
 Shigang Chen; Yi Deng; Attie, P.; Wei Sun;  
 Distributed Computing Systems, 1996., Proceedings of the 16th International Conferen  
 27-30 May 1996 Page(s):613 - 619  
[AbstractPlus](#) | Full Text: [PDF\(676 KB\)](#) IEEE CNF
- ☐ 4. **Efficient algorithms for detection and resolution of distributed deadlocks**  
 Shigang Chen; Yi Deng; Wei Sun; Rishe, N.;  
 Parallel and Distributed Processing, 1995. Proceedings. Seventh IEEE Symposium on  
 25-28 Oct. 1995 Page(s):10 - 16  
[AbstractPlus](#) | Full Text: [PDF\(556 KB\)](#) IEEE CNF
- ☐ 5. **SGTNE: semi-global time of the next event algorithm**  
 Boukerche, A.; Tropper, C.;  
 Parallel and Distributed Simulation, 1995. (PADS'95), Proceedings., Ninth Workshop o  
 No.95TB8096)  
 14-16 June 1995 Page(s):68 - 77  
[AbstractPlus](#) | Full Text: [PDF\(672 KB\)](#) IEEE CNF
- ☐ 6. **Hierarchical schedulings of time-next-event heuristic on distributed memory mac**  
 Boukerche, A.; Tropper, C.;  
 Simulation Symposium, 1995. Proceedings of the 28th Annual  
 9-13 April 1995 Page(s):155 - 165  
[AbstractPlus](#) | Full Text: [PDF\(720 KB\)](#) IEEE CNF



Terms used

[freeze](#) [list](#) or [snapshot](#) [sentence](#) [quiesce](#) [strength](#) or [level](#) [near/3](#) [risk](#) [near/3](#) [inconsisten](#) or [deadlock](#)

Found

1,795

157,87

Sort results  
by

relevance


[Save results to a Binder](#)

Try an [Advanced Search](#)

Display  
results

expanded form


[Search Tips](#)

Try this search in [The ACM Guide](#)

[Open results in a new window](#)

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)


Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐

## 1 [Fast detection of communication patterns in distributed executions](#)

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Full text available:  [pdf\(4.21 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

## 2 [Columns: Risks to the public in computers and related systems](#)

Peter G. Neumann

March 2004 **ACM SIGSOFT Software Engineering Notes**, Volume 29 Issue 2

Full text available:  [pdf\(165.39 KB\)](#)

Additional Information: [full citation](#)

## 3 [Papers from Hotnets-II: Unveiling the transport](#)

Jeffrey Mogul, Lawrence Brakmo, David E. Lowell, Dinesh Subhraveti, Justin Moore

January 2004 **ACM SIGCOMM Computer Communication Review**, Volume 34 Issue 1

Full text available:  [pdf\(120.97 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#)

Traditional application programming interfaces for transport protocols make a virtue of hiding most internal per-connection state. We argue that this information-hiding precludes many potentially useful application features and performance optimizations. We advocate a disciplined, portable, and secure interface that gives applications both "get" and "set" access to transport connection state.

## 4 [The family of concurrent logic programming languages](#)

Ehud Shapiro

September 1989 **ACM Computing Surveys (CSUR)**, Volume 21 Issue 3

Full text available:  [pdf\(9.62 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citing](#), [index terms](#)


Concurrent logic languages are high-level programming languages for parallel and distributed systems that offer a wide range of both known and novel concurrent programming techniques. Being logic programming languages, they preserve many advantages of the abstract logic programming model, including the logical reading of programs and computations, the convenience of representing data structures with logical terms and manipulating them using unification, and the amenability to metaprogrammin ...

## 5 [ARIES: a transaction recovery method supporting fine-granularity locking and partial](#)

rollbacks using write-ahead logging

C. Mohan, Don Haderle, Bruce Lindsay, Hamid Pirahesh, Peter Schwarz

March 1992 **ACM Transactions on Database Systems (TODS)**, Volume 17 Issue 1

Full text available:  [pdf\(5.23 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

DB2TM, IMS, and TandemTM systems. ARIES is applicable not only to database management systems but also to persistent object-oriented languages, recoverable file systems and transaction-based operating systems. ARIES has been implemented, to varying degrees, in IBM's OS/2TM Extended Edition Database Manager, DB2, Workstation Data Save Facility/VM, Starburst and QuickSilver, and in the University of Wisconsin's EXODUS and Gamma d ...

**Keywords:** buffer management, latching, locking, space management, write-ahead logging

6 Special issue: AI in engineering

D. Sriram, R. Joobhani

January 1985 **ACM SIGART Bulletin**, Issue 91

Full text available:  [pdf\(8.79 MB\)](#)

Additional Information: [full citation](#), [abstract](#)

The papers in this special issue were compiled from responses to the announcement in the July 1984 issue of the SIGART newsletter and notices posted over the ARPAnet. The interest being shown in this area is reflected in the sixty papers received from over six countries. About half the papers were received over the computer network.

7 The Kala basket: a semantic primitive unifying object transactions, access control, versions, and configurations

Sergui S. Simmel, Ivan Godard

November 1991 **ACM SIGPLAN Notices , Conference proceedings on Object-oriented programming systems, languages, and applications**, Volume 26 Issue 11


Full text available:  [pdf\(2.11 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

8 Multi-level transaction management for complex objects: implementation, performance, parallelism

Gerhard Weikum, Christof Hasse

October 1993 **The VLDB Journal – The International Journal on Very Large Data Bases**, Volume 2 Issue 4

Full text available:  [pdf\(2.83 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)


Multi-level transactions are a variant of open-nested transactions in which the subtransactions correspond to operations at different levels of a layered system architecture. They allow the exploitation of semantics of high-level operations to increase concurrency. As a consequence, undoing a transaction requires compensation of completed subtransactions. In addition, multi-level recovery methods must take into consideration that high-level operations are not necessarily atomic if multiple pages ...

**Keywords:** atomicity, complex objects, inter- and intratransaction parallelism, multi-level transactions, performance, persistence, recovery

9 A scalable formal method for design and automatic checking of user interfaces

Jean Berstel, Stefano Crespi Reghizzi, Gilles Roussel, Pierluigi San Pietro

April 2005 **ACM Transactions on Software Engineering and Methodology (TOSEM)**, Volume 14 Issue 2

Full text available:  [pdf\(1.74 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

The article addresses the formal specification, design and implementation of the behavioral component of graphical user interfaces. The complex sequences of visual events and actions that constitute dialogs are specified by means of modular, communicating grammars called VEG (Visual Event Grammars), which extend traditional BNF grammars to make them more convenient to model dialogs. A VEG specification is independent of the actual layout of the GUI, but it can easily be integrated with various la ...

**Keywords:** GUI design, Human-computer interaction (HCI), applications of model checking

10 Parallel execution of prolog programs: a survey

Gopal Gupta, Enrico Pontelli, Khayri A.M. Ali, Mats Carlsson, Manuel V. Hermenegildo  
July 2001 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 23 Issue 4

Full text available:  pdf(1.95 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Since the early days of logic programming, researchers in the field realized the potential for exploitation of parallelism present in the execution of logic programs. Their high-level nature, the presence of nondeterminism, and their referential transparency, among other characteristics, make logic programs interesting candidates for obtaining speedups through parallel execution. At the same time, the fact that the typical applications of logic programming frequently involve irregular computatio ...

**Keywords:** Automatic parallelization, constraint programming, logic programming, parallelism, prolog

11 A critique of ANSI SQL isolation levels

Hal Berenson, Phil Bernstein, Jim Gray, Jim Melton, Elizabeth O'Neil, Patrick O'Neil  
May 1995 **ACM SIGMOD Record , Proceedings of the 1995 ACM SIGMOD international conference on Management of data**, Volume 24 Issue 2

Full text available:  pdf(1.20 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

ANSI SQL-92 [MS, ANSI] defines Isolation *Levels* in terms of *phenomena*: Dirty Reads, Non-Repeatable Reads, and Phantoms. This paper shows that these phenomena and the ANSI SQL definitions fail to properly characterize several popular isolation levels, including the standard locking implementations of the levels covered. Ambiguity in the statement of the phenomena is investigated and a more formal statement is arrived at; in addition new phenomena that better characterize isolation t ...

12 Illustrative risks to the public in the use of computer systems and related technology

Peter G. Neumann  
January 1996 **ACM SIGSOFT Software Engineering Notes**, Volume 21 Issue 1

Full text available:  pdf(2.54 MB) Additional Information: [full citation](#)

13 Technical reports

SIGACT News Staff  
January 1980 **ACM SIGACT News**, Volume 12 Issue 1

Full text available:  pdf(5.28 MB) Additional Information: [full citation](#)

14 Special issue: Game-playing programs: theory and practice

M. A. Bramer  
April 1972 **ACM SIGART Bulletin**, Issue 80

Full text available:  pdf(9.23 MB) Additional Information: [full citation](#), [abstract](#)

This collection of articles has been brought together to provide SIGART members with an overview of Artificial Intelligence approaches to constructing game-playing programs. Papers on both theory and practice are included.

15 System R: relational approach to database management

M. M. Astrahan, M. W. Blasgen, D. D. Chamberlin, K. P. Eswaran, J. N. Gray, P. P. Griffiths, W. F. King, R. A. Lorie, P. R. McJones, J. W. Mehl, G. R. Putzolu, I. L. Traiger, B. W. Wade, V. Watson  
June 1976 **ACM Transactions on Database Systems (TODS)**, Volume 1 Issue 2

Full text available:  pdf(3.18 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

System R is a database management system which provides a high level relational data interface. The systems provides a high level of data independence by isolating the end user as much as possible from underlying storage structures. The system permits definition of a variety of relational views on common underlying data. Data control features are provided, including authorization, integrity assertions, triggered transactions, a logging and recovery subsystem, and facilities for maintaining ...

**Keywords:** authorization, data structures, database, index structures, locking, nonprocedural language, recovery, relational model

16 Research session 5: data mining / transaction management: Allocating isolation levels to transactions □

Alan Fekete

June 2005 **Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems**

Full text available:  [pdf\(227.00 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Serializability is a key property for executions of OLTP systems; without this, integrity constraints on the data can be violated due to concurrent activity. Serializability can be guaranteed regardless of application logic, by using a serializable concurrency control mechanism such as strict two-phase locking (S2PL); however the reduction in concurrency from this is often too great, and so a DBMS offers the DBA the opportunity to use different concurrency control mechanisms for some transaction ...

**Keywords:** anomaly, concurrency control, consistency, serializability, snapshot isolation, two-phase locking

17 Data base directions: the next steps □

John L. Berg

November 1976 **ACM SIGMOD Record , ACM SIGMIS Database**, Volume 8 , 8 Issue 4 , 2

Full text available:  [pdf\(9.95 MB\)](#) Additional Information: [full citation](#), [abstract](#)

What information about data base technology does a manager need to make prudent decisions about using this new technology? To provide this information the National Bureau of Standards and the Association for Computing Machinery established a workshop of approximately 80 experts in five major subject areas. The five subject areas were auditing, evolving technology, government regulations, standards, and user experience. Each area prepared a report contained in these proceedings. The proceedings p ...

**Keywords:** DBMS, auditing, cost/benefit analysis, data base, data base management, government regulation, management objectives, privacy, security, standards, technology assessment, user experience

18 The evolution of Coda □

M. Satyanarayanan

May 2002 **ACM Transactions on Computer Systems (TOCS)**, Volume 20 Issue 2

Full text available:  [pdf\(441.35 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Failure-resilient, scalable, and secure read-write access to shared information by mobile and static users over wireless and wired networks is a fundamental computing challenge. In this article, we describe how the Coda file system has evolved to meet this challenge through the development of mechanisms for server replication, disconnected operation, adaptive use of weak connectivity, isolation-only transactions, translucent caching, and opportunistic exploitation of hardware surrogates. For eac ...

**Keywords:** Adaptation, Linux, UNIX, Windows, caching, conflict resolution, continuous data access, data staging, disaster recovery, disconnected operation, failure, high availability, hoarding, intermittent networks, isolation-only transactions, low-bandwidth networks, mobile computing, optimistic replica control, server replication, translucent cache management, weakly connected operation

- 19 POESIA: An ontological workflow approach for composing Web services in agriculture [  
Renato Fileto, Ling Liu, Calton Pu, Eduardo Delgado Assad, Claudia Bauzer Medeiros  
November 2003 **The VLDB Journal – The International Journal on Very Large Data Bases**,  
Volume 12 Issue 4

Full text available:  pdf(726.49 KB) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

This paper describes the POESIA approach to systematic composition of Web services. This pragmatic approach is strongly centered in the use of domain-specific multidimensional ontologies. Inspired by applications needs and founded on ontologies, workflows, and activity models, POESIA provides well-defined operations (aggregation, specialization, and instantiation) to support the composition of Web services. POESIA complements current proposals for Web services definition and composition by provi ...

**Keywords:** Composition of Web services, Data integration, Ontologies, Semantic Web, Semantics of data and processes

- 20 Converging CSP specifications and C&plus;&plus; programming via selective formalism [  
William B. Gardner  
May 2005 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 4 Issue 2

Full text available:  pdf(617.07 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

CSP (communicating sequential processes) is a useful algebraic notation for creating a hierarchical behavioral specification for concurrent systems, due to its formal interprocess synchronization and communication semantics. CSP specifications are amenable to simulation and formal verification by model-checking tools. A translator has been created to synthesize C&plus;&plus; code from CSP for execution with an object-oriented framework called CSP&plus;&plus;;, thereby making CSP specifications di ...

**Keywords:** Executable specifications, hardware/software codesign, object-oriented application frameworks

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.  
[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)